

I hereby certify that this correspondence is being electronically transmitted to the United States Patent and Trademark Office on August 11, 2009.


Sarah J. Knight, Patent Attorney

REQUEST FOR CERTIFICATE
OF CORRECTION UNDER
37 CFR 1.322 and 1.323
Docket No. UF.318X
Patent No. 7,509,300 B2

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Applicants : Sartaj Kumar Sahni, Haibin Lu
Issued : March 24, 2009
Patent No. : 7,509,300 B2
Conf. No. : 6767
For : Dynamic IP Router Tables Using Highest-Priority Matching

ATTN: CERTIFICATES OF CORRECTION BRANCH
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

REQUEST FOR CERTIFICATE OF CORRECTION
UNDER 37 CFR 1.322 (OFFICE MISTAKE) AND
UNDER 37 CFR 1.323 (APPLICANTS' MISTAKE)

Sir:

A Certificate of Correction for the above-identified patent has been prepared and is attached hereto.

In the left-hand column below is the column and line number where errors occurred in the patent. In the right-hand column is the page and line number in the application where the correct information appears.

Patent Reads:

Column 6, Line 22:
"point(root)≤start(r)"

Patent Reads:

Column 8, Line 14:
"than a new node"

Application Reads:

Page 9, Lines 5-6:
--point(root) < start(r)--

Application Should Read:

Page 11, Line 34:
--then a new node--

Patent Reads:

Column 8, Line 64:
"rotation ranges(z)"

Column 8, Line 66:
"rotation ranges(subtree(x))"

Patent Reads:

Column 10, Line 42:
"a router tables"

Application Reads:

Page 13, Line 5:
--rotation. ranges(z)--

Page 13, Line 7:
--rotation. ranges(subtree(x))--

Application Should Read:

Page 15, Line 25:
--a router table--

A true and correct copy of pages 9 and 13 of the specification as filed, which supports Applicants' assertion of errors on the part of the Patent Office, accompanies this Certificate of Correction.

The Commissioner is authorized to charge the fee of \$100.00 for the amendment to Deposit Account No. 19-0065. The Commissioner is also authorized to charge any additional fees as required under 37 CFR 1.20(a) to Deposit Account No. 19-0065.

Approval of the Certificate of Correction is respectfully requested.

Respectfully submitted, -



Sarah J. Knight
Patent Attorney
Registration No. 58,722
Phone No.: 352-375-8100
Fax No.: 352-372-5800
Address: P.O. Box 142950
Gainesville, FL 32614-2950

SJK/jlr

Attachments: Specification as filed, pages 9 and 13
Certificate of Correction

$point(z)$, and nodes in the right subtree of z have larger point values than $point(z)$. Where R is the set of nonintersecting ranges of the NHRT, each range of R is stored in exactly one of the nodes of the PTST. Thus, the root of the PTST stores (1) all ranges $r \in R$ such that $start(r) \leq point(root) \leq finish(r)$; (2) all ranges $r \in R$ such that $finish(r) < point(root)$ are stored in the left subtree of the root; and (3) all ranges $r \in R$ such that $point(root) < start(r)$ (i.e., the remaining ranges of R) are stored in the right subtree of the root, hereinafter referred to as the “range allocation rule.” The ranges allocated to the left and right subtrees of the root are allocated to nodes in these subtrees using the range allocation rule recursively.

Referring now to Figures 1A and 1B, a PTST is created and assigned node point values using the range allocation rule, wherein nodes in the left subtree have smaller point values than the root node, and nodes in the right subtree have larger point values than the root node. Next, the nonintersecting ranges are allocated to the nodes of the newly created PTST, wherein all ranges containing the point value of the root node are stored in the root node. Further, all ranges having a finish point less than the chosen root node are stored in the left subtree of the root node, and all ranges having a start point greater than the chosen root node are stored in the right subtree of the root node. The PTST is then populated with the allocated ranges and corresponding priorities. The following Table 1 provides an example set of non-intersecting ranges and Figure 1B illustrates a possible PTST for the set of ranges provided in Table 1.

Table 1—A non-intersecting range set

Range	Priority
[2, 100]	4
[2, 4]	33
[2, 3]	34
[8, 68]	10
[8, 50]	9
[10, 50]	20
[10, 35]	3
[15, 33]	5
[16, 30]	30
[54, 66]	18
[60, 65]	7
[69, 72]	10
[80, 80]	12

Using the following lemma (and corresponding proof), a red-black tree can be rebalanced using LL or RR rotations.

Lemma: R is a set of nonintersecting ranges and $ranges(z) \subseteq R$ are the ranges allocated by the range allocation rule to node z of the PTST prior to an LL or RR rotation.

- 5 Let $ranges'(z)$ be this subset for the PTST node z following the rotation. $ranges(z) = ranges'(z)$ for all nodes z in the subtrees a , b , and c of Figures 3B and 3C.

Proof: consider an LL rotation. $ranges(subtree(x))$ is the union of the ranges allocated to the nodes in the subtree whose root is x . Thus, the range allocation rule allocates each range r to the node z nearest the root such that r matches $point(z)$,
 10 $ranges(subtree(x)) = ranges'(subtree(y))$. Further, $r \in ranges(a)$ if $r \in ranges(subtree(x))$ and $finish(r) < point(y)$. Consequently, $r \in ranges'(a)$. From this and the fact that the LL rotation does not change the positioning of nodes in a , it follows that for every node z in the subtree a , $ranges(a) = ranges'(a)$. The proof for the nodes in b and c as well as for the RR rotation is similar.

- 15 With reference to Figures 3B and 3C, it follows from the previously described Lemma that $ranges(z) = ranges'(z)$ for all z in the PTST except possibly for $z \in \{x, y\}$. Thus, $ranges'(y) = ranges(y) \cup S$ and $ranges'(x) = ranges(x) - S$, where $S = \{r | r \in ranges(x) \wedge start(r) \leq point(y) \leq finish(r)\}$. Because the ranges are nonintersecting ranges, all ranges in $ranges(y)$ are nested within the ranges of S . In addition, the range
 20 $rMax$ of S with largest $start()$ value may be found by searching $RST(x)$ for the range with largest $start()$ value that matches $point(y)$. Where $RST(x)$ is a binary search tree of an ordered set of ranges, $rMax$ may be found in $O(height(RST(x)))$ time by following a path from the root downward. Where $rMax$ does not exist, $S = \emptyset$, $ranges'(x) = ranges(x)$ and $ranges'(y) = ranges(y)$.

- 25 With the assumption that $rMax$ exists, the operation may be "split" to extract from $RST(x)$ the ranges that belong to S . The following split operation: $RST(x) \rightarrow split(small, rMax, big)$ separates $RST(x)$ into an RST $small$ of ranges $<$ than $rMax$ and an RST big of ranges $>$ than $rMax$. Thus, $RST'(x) = big$ and $RST'(y) = join(small, rMax, RST(y))$, where "join" (see Horowitz *et al.*, Fundamentals of Data Structures in C++, W.H
 30 Freeman, NY, 653 pages (1995)) combines the red-black tree $small$ with ranges $< rMax$,

UNITED STATES PATENT AND TRADEMARK OFFICE

CERTIFICATE OF CORRECTION

PATENT NO. : 7,509,300 B2

Page 1 of 1

APPLICATION NO.: 10/613,963

DATED : March 24, 2009

INVENTORS : Sartaj Kumar Sahni, Haibin Lu

It is certified that errors appear in the above-identified patent and that said Letters Patent is hereby corrected as shown below:

Column 6,

Line 22, "point(root) \leq start(r)" should read --point(root) < start(r)--.

Column 8,

Line 14, "than a new node" should read --then a new node--.

Line 64, "rotation ranges(z)" should read --rotation. ranges(z)--.

Line 66, "rotation ranges(subtree(x))" should read --rotation. ranges(subtree(x))--.

Column 10,

Line 42, "a router tables" should read --a router table--.

MAILING ADDRESS OF SENDER:

Saliwanchik, Lloyd & Saliwanchik

P.O. Box 142950

Gainesville, FL 32614-2950